



Høgskolen i Telemark

EKSAMEN

I

5608 PROGRAMMERING

04.05. 2010

Tid:	5 timar, 09:00 – 14:00
Målform:	Bokmål / Nynorsk
Sidetal:	6 med framsida
Hjelpemiddel:	Alt skriftleg materiale
Merknadar:	Vekting er gitt ved starten av kvar oppgåve.
Vedlegg:	Ingen

Eksamensresultata blir offentleggjort på studentweb.



BOKMÅL

Oppgave 1**Vekting: 20%**

- 1 - a) Forklar kort hva som blir skrevet ut når programmet Oppgave1a blir kjørt på en maskin.

```
public class Oppgave1a {
    public static void main(String[] args) {
        int t = 373;
        String u = t + ": ";
        if ( t%10 == t/100 )
            u += "s";
        else
            u += "u";
        System.out.println(u);
    }
}
```

- 1 - b) Koden nedenfor er et utsnitt hentet fra en metode der g er et Graphics-objekt. Skisser resultatet som fremkommer ved bruk av akkurat disse instruksjonene:

```
:
int x = 50, y = 50;
int f = 100;
g.drawRect(x, y, 2*f, 2*f);
g.drawOval(x, y, f, f);
g.drawRect(x+f, y, f, f);
g.drawRect(x, y+f, f, f);
g.drawOval(x+f, y+f, f, f);
:
```

- 1 - c) Skriv kun de instruksjonene (ikke et komplett program) som er nødvendig for å sjekke om innholdet i String-variabelen spesOrd har følgende egenskap: Dersom første bokstav tas ut av et ord og så settes inn igjen bakerst i resten av ordet, så får man det opprinnelige ordet når det nye leses baklengs.

For eksempel: solo --> olos og irettesetter --> rettesetteri
men ikke: kasse --> assek eller helge --> elgeh

Oppgave 2**Vekting: 10%**

Skriv et komplett javaprogram som ber brukeren om å gi inn størrelsen på en vinkelfigur som deretter skal tegnes ut. Figuren skal være en slags L (se figurene under), der innlest størrelse svarer til lengden på hver sidekant, tegnet ved hjelp av X-er i et dialogvindu.

Vinkelfigur, størrelse 3:

```
X
X
X X X
```

Vinkelfigur, størrelse 4:

```
X
X
X
X X X X
```

Vinkelfigur, størrelse 5:

```
X
X
X
X
X
X X X X X
```

Oppgave 3**Vekting: 30%**

Ola bruker skrittelleren sin systematisk. Hver dag etter jobben leser han av antall skritt han har gått i arbeidssammenheng, og når han går for å legge seg leser han av antall skritt han har gått på fritiden. Ola registrerer de to skrittantallene på en sekvensiell tekstfil, en linje pr dag og en fil for hver måned. Merk at 0 skritt på jobb betyr at Ola har hatt fri den dagen (f.eks. påskeferie), mens en eventuell 0 på fritida betyr at han har vært syk den dagen (sengeliggende).

Etter hver måned ønsker Ola å kunne bestemme det gjennomsnittlige antallet skritt som er gått henholdsvis på jobb og på fritiden i den måneden. Ola vil ta ut av den aktuelle snittberegningen de dagene det eventuelt er gått 0 skritt.

```

april.txt
1 - 0 - 11010
2 - 0 - 13235
:
:
29 - 7130 - 6685
30 - 9245 - 3850
  
```

På figuren til venstre ser vi at hver linje på filen inneholder dagnummer, etterfulgt av antall skritt på jobb og antall skritt på fritiden.

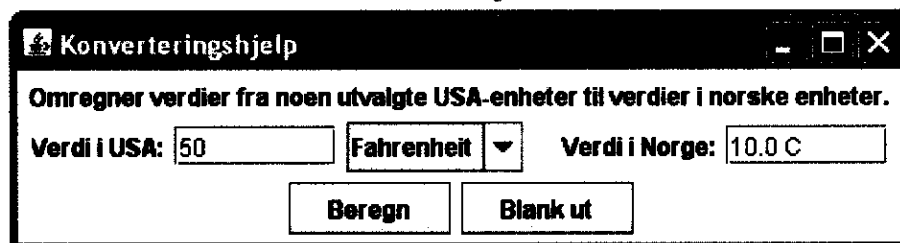
Ola bruker bindestrek og mellomrom for å skille mellom de to registrerte skrittantallene.

Skriv et komplett javaprogram som kan hjelpe Ola i arbeidet med å beregne de to snittverdiene. Programmet skal først be om og lese inn navnet på en av de månedlige registreringsfilene, så beregne de to gjennomsnittlige skrittantallene og deretter skrive disse inn på slutten av den samme filen.

Oppgave 4**Vekting: 30%**

I USA bruker de noen andre enheter enn det vi gjør her i Norge, for eksempel mynt - *dollar*, temperatur - *Fahrenheit* og lengde - *mile*. Skriv et komplett java-program som basert på et grafisk brukergrensesnitt (GUI), kan brukes til omregning fra en av disse tre nevnte enhetene i USA til de tilsvarende verdiene i Norge. Verdi som skal omregnes skrives inn i et tekstfelt, den aktuelle enheten velges i en nedtrekksliste og etter trykk på en beregningsknapp så skal den tilsvarende verdien i norske enheter vises i et eget tekstfelt. Det skal også være en knapp for å kunne starte en ny omregning.

(450 x 120 pix)



Eksempel på et GUI for enhetsomregning

Bruk følgende sammenhenger:

1 dollar = 5.87 kr

$C = (5/9) * (F - 32)$, der C er Celsiusgrader og F er Fahrenheitgrader

1 mile = 1.6093 km

Oppgave 5**Vekting: 10%**

Et program skal benytte en ferdig utviklet klasse `Mynt`, som vi ikke trenger å kjenne i detalj, men det er gitt at denne klassen er sorterbar (*implements Comparable*) etter antatt salgsverdi (ikke pålydende verdi).

`Mynt`-klassen skal brukes til å håndtere en allerede registrert myntsamling som er lagt inn i tabellen `myntListe`. Skriv kun de instruksjonene som er nødvendige for at et slikt program skal kunne skrive ut alle registrerte opplysninger om de 10 mest verdifulle myntene i denne samlingen i et dialogvindu.

Du skal altså ikke skrive selve klassen `Mynt`, bare bruke den. Kommenter hvilke forutsetninger som gjelder for de instruksjonene som blir satt opp.

SLUTT

 NYNORSK

Oppgave 1**Vekting: 20%**

- 1 - b) Forklar kort kva som blir skrevet ut når programmet `Oppgaaave1a` blir kjørt på ein maskin.

```
public class Oppgaaave1a {
    public static void main(String[] args) {
        int t = 373;
        String u = t + ": ";
        if ( t%10 == t/100 )
            u += "s";
        else
            u += "u";
        System.out.println(u);
    }
}
```

- 1 - d) Koden nedanfor er eit utsnitt henta frå ein metode der `g` er eit `Graphics`-objekt. Skisser resultatet som kjem fram ved bruk av akkurat desse instruksjonane:

```
:
int x = 50, y = 50;
int f = 100;
g.drawRect(x, y, 2*f, 2*f);
g.drawOval(x, y, f, f);
g.drawRect(x+f, y, f, f);
g.drawRect(x, y+f, f, f);
g.drawOval(x+f, y+f, f, f);
:
```

- 1 - e) Skriv kun dei instruksjonane (ikkje eit komplett program) som er naudsynte for å sjekke om innhaldet i `String`-variabelen `spesOrd` har følgjande eigenskap: Dersom første bokstav blir teken ut av eit ord og så sett inn att bakerst i resten av ordet, så får ein det opprinnelege ordet når det nye lesast baklengs.

Til dømes: `solo` --> `olos` og `irettesetter` --> `rettesetteri`
 men ikkje: `kasse` --> `assek` eller `helge` --> `elgeh`

Oppgave 2**Vekting: 10%**

Skriv eit komplett javaprogram som ber brukaren om å gi inn storleiken på ein vinkelfigur som deretter skal teknast ut. Figuren skal vere ein slags L (sjå figurane under), der innlest storleik svarer til lengda på kvar sidekant, teikna ved hjelp av X-ar i eit dialogvindu.

Vinkelfigur, storleik 3:

```
X
X
X X X
```

Vinkelfigur, storleik 4:

```
X
X
X
X X X X
```

Vinkelfigur, storleik 5:

```
X
X
X
X
X
X X X X X
```

Oppgave 3**Vekting: 30%**

Ola bruker skritteljaren sin systematisk. Kvar dag etter jobben les han av talet på skritt han har gått i arbeidssamanheng, og når han går for å leggje seg les han av talet på skritt han har gått på fritida. Ola registrerer dei to skritt-tala på ei sekvensiell tekstfil, ei line pr dag og ei fil for kvar månad. Merk at 0 skritt på jobb tyder at Ola har hatt fri den dagen (f.eks. påskeferie), medan ein eventuell 0 på fritida tyder at han har vore sjuk den dagen (sengjeliggande).

Etter kvar månad ønskjer Ola å kunne bestemme det gjennomsnittlege talet på skritt som er gått høvesvis på jobb og på fritida i den månaden. Ola vil ta ut av den aktuelle snittberekninga dei dagane det eventuelt er gått 0 skritt.

april.txt		
1	- 0 -	11010
2	- 0 -	13235
:		
:		
29	- 7130 -	6685
30	- 9245 -	3850

På figuren til venstre ser vi at kvar line på fila inneheld dagnummer, etterfølgd av talet på skritt på jobb og talet på skritt på fritida.

Ola bruker bindestrek og mellomrom for å skilje mellom dei to registrerte skritt-tala.

Skriv eit komplett javaprogram som kan hjelpe Ola i arbeidet med å rekne ut dei to snittverdiane. Programmet skal først be om og lese inn navnet på ei av dei månadlege registreringsfilene, så berekne dei to gjennomsnittlege skritt-tala og deretter skrive desse inn på slutten av den same fila.

Oppgave 4**Vekting: 30%**

I USA bruker dei nokre andre einingar enn det vi gjer her i Noreg, for eksempel mynt - *dollar*, temperatur - *Fahrenheit* og lengde - *mile*. Skriv eit komplett javaprogram som basert på eit grafisk brukargrensesnitt (GUI), kan nyttast til omrekning frå ein av desse tre einingane i USA til dei tilsvarande verdiane i Noreg. Verdi som skal omreknast skrivast inn i eit tekstfelt, den aktuelle eininga blir valt i ei nedtrekksliste og etter trykk på ein omrekningsknapp så skal den tilsvarande verdien i norsk eining visast i eit eige tekstfelt. Det skal og vere ein knapp for å kunne starte ei ny omrekning. Sjå figuren på neste side.

(450 x 120 pix)

Eksempel på eit GUI for omrekning mellom ulike einingar

Bruk følgjande samanhengar:

1 dollar = 5.87 kr

$C = (5/9) * (F - 32)$, der C er Celsiusgrader og F er Fahrenheitgrader

1 mile = 1.6093 km

Oppgåve 5

Vekting: 10%

Eit program skal bruke ein ferdig utvikla klasse `Mynt`, som vi ikkje treng å kjenne i detalj, men det er gitt at denne klassen er sorterbar (*implements Comparable*) etter anteken salsverdi (ikkje pålydande verdi).

`Mynt`-klassen skal brukast til å handtere ei allereie registrert myntsamling som er lagt inn i tabellen `myntListe`. Skriv kun dei instruksjonane som er naudsynt for at eit slikt program skal kunne skrive ut alle registrerte opplysningar om dei 10 mest verdfulle myntane i denne samlinga i eit dialogvindu.

Du skal altså ikkje skrive sjølve klassen `Mynt`, bare bruke han. Kommenter kva for forutsetningar som gjeld for dei instruksjonane som blir sett opp.

SLUTT